מחשבת 2

תיכנות לשעות הפנאי

commodore64

משהו על חברת "מחשבת" – מערכות למידה

זו חברה, אשר אחד מתחומי התמחותה העיקריים הוא פיתוח מערכות לימוד עצמי.
מחברי יחידות לימוד אלה עוסקים במשך שנים בפיתוח מערכות ללימוד עצמי עבור המגזר האזרחי והצבאי.

ת.ד. 48032 ת"א מיקוד 61480.
הוצאה לאור – מחשבת מ.ל. בע"מ.

commodore 64
vic 20



מחשבת

צעדים ראשונים במחשב
בלימוד עצמי

יחידה 2
ריבועי־קסם

תוכן העניינים

פתח דבר

זוכר את השורה בעזרתה קיבלת פסק זמן?

FOR T=1 TO 1ØØØ : NEXT

שורה זאת איפשרה לרובוט הממוחשב "איציק" לבצע התעמלות בקצב סביר.

ביחידת לימוד זאת "נפצח" ונבין את המסתתר מאחורי הפקודות המופיעות בשורה זאת ונראה כי יש להם מקום מרכזי בכל תוכנית בשפת ה-BASIC.

שמנו לעצמנו כמטרה ללמד ביחידת לימוד זו את המושגים הבסיסיים והחשובים ביותר בשפת הבייסיק:

- לולאות סופיות (FOR-NEXT).
- משתנים מספריים ומחרוזתיים.
- פקודות תנאי (IF ... THEN).

נראה כיצד מושגים אלה יפתחו בפנינו עולם ומלואו:
החל בבנית שעון דיגיטלי, דרך בנית ריבועי קסם וכלה בפיתוח משחק מחשב-טלויזיה.

באיחולי המשך
לימוד מהנה ומרתק
דני קדם ויצחק קליסקי
"מחשבת" - מערכות למידה

פרק א
המחשב כמכונת שכפול

לולאות FOR-NEXT

האם אתה זוכר את התוכנית:

```
10  PRINT  "ישמך"
20  GO TO 10
```

מיחידה 1?

הפקודה GO TO המופיעה בשורה 20 גרמה לכך שהמחשב הסתובב בלולאה אינסופית וביצע פעם אחר פעם את שורה 10. פעולה זו נפסקה רק כאשר לחצת על מקש RUN-STOP.

בעזרת ההוראות FOR-NEXT תוכל להכתיב למחשב כמה פעמים לבצע כל הוראה:

? האם תוכל לנחש כמה פעמים ירשום המחשב את שמך בתוכנית הבאה?

```
10 FOR N=1 TO 5
20 PRINT "ישמך"
30 NEXT N
```

■ העתק את התוכנית למחשב והרץ אותה!
(לפני כן נקה את הזכרון מכל תוכנית קודמת וכן את המסך.)

אתה רואה כי שמך הופיע 5 פעמים זה מתחת לזה.

■ שנה את התוכנית כך, שהמחשב ישכפל את שמך 8 פעמים.
(תשובה 1 בעמוד 56)

נסכם:

כאשר אתה כותב תוכנית עם צמד הפקודות FOR-NEXT אתה יכול לשלוט במספר הפעמים שהמחשב מבצע את השורות הכתובות בין צמד פקודות זה.

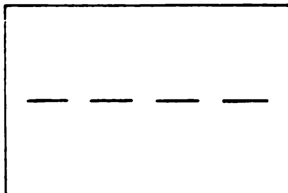
למשל:

```
10 FOR N=1 TO 20
20 .............
30 .............
40 .............
50 NEXT N
```

? כמה פעמים יבצע המחשב את שורות 20, 30 ו-40?
(תשובה 2 בעמוד 56)

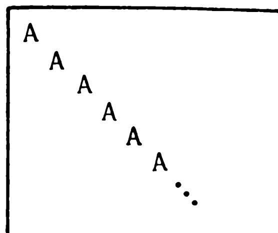
■ כתוב תוכנית שתדפיס את שמך 4 פעמים במרכז המסך:

שים לב: עליך לחשוב כל הזמן היכן נמצא המסמן בסוף כל סיבוב בלולאה, ואיך להביא אותו למקום הרצוי על המסך לקראת תחילת הסיבוב הבא בלולאה.



(תשובה 3 בעמוד 56)

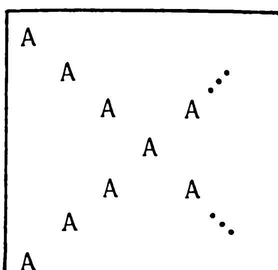
■ כתוב תוכנית אשר תדפיס 21 פעמים את האות A באלכסון על המסך החל מהפינה השמאלית-עליונה:
(השתמש ב-FOR-NEXT)



(תשובה 4 בעמוד 56)

■ הוסף עוד שורות לתוכנית כך, שהמחשב ידפיס עוד 21 פעמים A באלכסון השני:

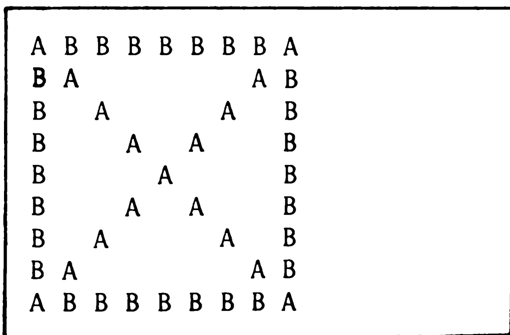
<u>הערה</u>: אם ה-READY המודפס בתום ריצת התוכנית מפריע לך - העלם אותו!



(תשובה 5 בעמוד 56)

משימה:

השלם את התוכנית כך שתדפיס על המסך את הצורה הבאה:

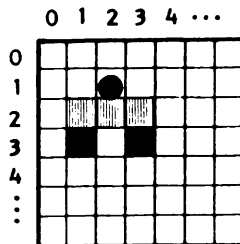


הפעם אין אנו נותנים תשובה אלא מצפים כי תתמודד בעצמך עם כל הבגים.

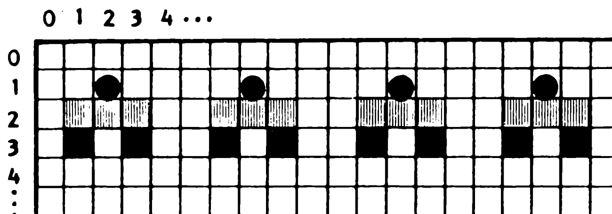
שכפול רובוטים

עכשיו נשתמש בצמד הפקודות FOR-NEXT על-מנת לשכפל צורות מורכבות על המסך.

הכר בבקשה את אי-לי, בן דודו של אי-ציק מהיחידה הקודמת:



כתוב תוכנית (עם FOR-NEXT) כך שאי-לי יופיע 4 פעמים לרוחב המסך:



מתקשה לכתוב את התוכנית?

הנה כמה רמזים:

- צייר תחילה את אי-לי פעם אחת על המסך.
- חשוב היכן נמצא המסמן בגמר השרטוט של אי-לי והכנס פקודות כאלה שיביאו את המסמן למקום בו אתה רוצה להתחיל את הרובוט השני.
- "עטוף" את התוכנית בצמד פקודות FOR-NEXT כך, שאי-לי ישוכפל 4 פעמים.

לאחר שהתמודדת עם הבעיה פנה לתשובה 6 בעמוד 57

שכלולים

- שכפל את אי-לי באלכסון.
- שכפל אותו בשני האלכסונים.
- צבע את אי-לי בשלל צבעים.

פרק ב
משתנים ולולאות

תוצאה מפתיעה

? האם אתה יכול לשער מה יתקבל על המסך לאחר שתריץ את התוכנית?

```
10 FOR N=1 TO 10
20 PRINT N      ← אל תדפיס מרכאות
30 NEXT
```

■ העתק את התוכנית למחשב והרץ אותה.

אם פעלת נכון, הרי שהמחשב רשם על המסך אחד מתחת לשני את המספרים מ-1 עד 10.

כיצד קרה הדבר?

משתנים

כדי להבין זאת עליך להכיר מהו **משתנה**.

בשלב זה מחק את התוכנית מזכרונו של המחשב ועבור לשלב הבא:

■ פקוד על המחשב:

```
LET A=6
LET B=4       ולאחר מכן
```

(אל תשכח ללחוץ RETURN אחרי כל פקודה)

■ נקה את כל המסך.

? מה יקרה עכשיו, אם תפקוד על המחשב:

```
PRINT A + B
```

■ חשוב תחילה ואח״כ נסה! (אל תשכח כמובן ללחוץ על RETURN)

? המחשב הדפיס את המספר: _____ (0, 6, 4, 10)

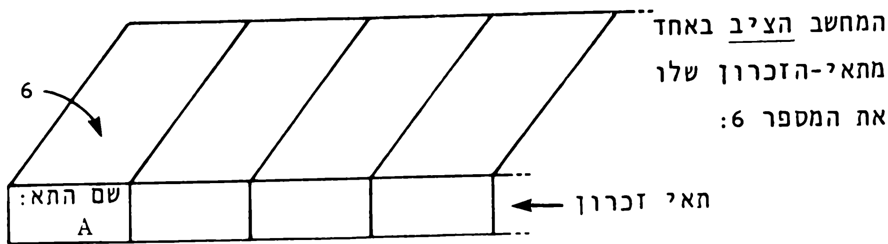
כיצד קרה הדבר?

ובכן, כאשר פקדת על המחשב:

LET A=6

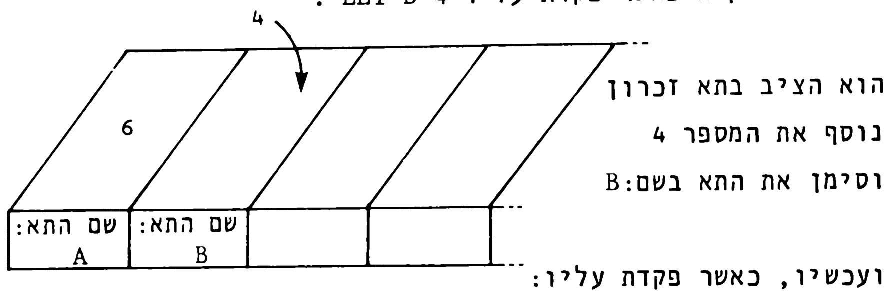
הוא ביצע את הפעולה המעניינת הבאה:

המחשב הציב באחד מתאי-הזכרון שלו את המספר 6:



וסימן את התא שלתוכו הוא הכניס את 6 בשם:A

דבר דומה קרה כאשר פקדת עליו LET B=4 :



הוא הציב בתא זכרון נוסף את המספר 4 וסימן את התא בשם:B

ועכשיו, כאשר פקדת עליו:

PRINT A+B

הוא ביצע את הפעולות הבאות:

- חיפש תא בשם - A, מצא אותו, והעתיק את המספר שנמצא בתוכו (6) אל מרכז החישובים הנמצא ב"מוחו".
- המחשב המשיך וחיפש תא בשם - B, מצא אותו, והעתיק את המספר שהיה בו אל אותו מרכז חישובים.
- מרכז חישובים זה נקרא בשפה מקצועית: היחידה האריתמטית. ביחידה האריתמטית הוא חיבר את 6 ואת 4 והציג את התוצאה על המסך.

? האם לאחר שהמחשב ביצע את פעולת החיבור עדיין נמצאים המספרים 6 ו-4 בתאים A ו-B?
בדוק זאת על המחשב:

■ פקוד עליו להדפיס את ערכו של A (PRINT A).

? ומה לגבי ערכו של B?

מסקנה: כדי לבצע פעולות ב-A וב-B <u>העתיק</u> המחשב את ערכם מן התאים שלהם. הערך של A ו-B נשאר בזכרון!

? מה יתקבל על המסך אם תפקוד על המחשב:

```
PRINT A*B
```

(להזכיר: * - הוא סימן הכפל במחשב).

■ פקוד על המחשב לבצע את פעולת הכפל ובדוק אם צדקת.

■ עכשיו פקוד על המחשב:

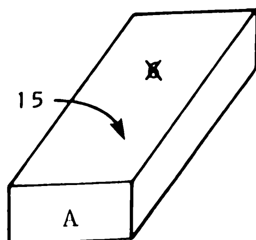
```
LET A = 15
```

? מה ידפיס המחשב לאחר שתפקוד עליו PRINT A: __________
(5, 6, 15, 15 ו-6 ביחד)

■ פקוד עליו להדפיס את A! האם צדקת?

הסבר:

- כאשר פקדת מקודם LET A=6 הכניס המחשב לתא זכרון את המספר 6 וקרא לתא - A.
- כאשר פקדת אח"כ LET A=15 חיפש המחשב בזכרונו לראות אם קיים תא בשם A. הוא, כמובן, מצא תא כזה ואז הכניס פנימה את המספר 15, ומחק את המספר 6:



נוכחת לדעת כי A יכול לייצג מספרים כלשהם בהתאם למה שנמצא בתא הזכרון A.

מסיבה זו, "יצורי" כמו A נקרא בשם **משתנה**.

(ליתר דיוק - משתנה מספרי)

■ פקוד על המחשב לתת למשתנה B את הערך 5.

? מה ידפיס המחשב כאשר תפקוד עליו:

```
PRINT A/B
PRINT 3*A
```

■ נסה!

? מה יעשה המחשב כאשר תפקוד עליו:

```
LET C = A + B
```

במקרה זה הגדרנו משתנה חדש, C, אשר שווה לסכום שני המשתנים A ו-B. (בדוק זאת ע"י PRINT C)

? בדוק מה ידפיס המחשב במקרה הבא:

```
PRINT D
```

הסבר:

המחשב בודק אם קיים בזכרונו משתנה בשם D. כמובן שהוא לא מצא משתנה בשם זה ולכן D קיבל את הערך Ø.

? מה יקרה כאשר תפקוד:

```
PRINT A/F
```

■ נסה! מדוע מודיע לך המחשב שניסית לחלק באפס?

קיצורי־דרך בכתיבה

עד כה הגדרנו משתנים בעזרת פקודת LET.
מחשב הקומודור שלך יודע לעשות זאת גם אם תוותר על כתיבת המילה LET:

■ נסה לכתוב

A = 5Ø

■ בקש מן המחשב להדפיס את ערכו של A.

שמות של משתנים

איזה שם מותר לתת למשתנה?
השם שהמחשב נותן לתא בזכרון בו נמצא הערך של המשתנה, חייב להתחיל באות אנגלית. אחרי אות זו יכולה לבוא עוד אות אנגלית או עוד ספרה אחת.

? אילו מבין שמות המשתנים הבאים איננו חוקי? :

ZX, B, 1C, BC, 2D, 25, XY

(תשובה 7 בעמוד 57)

■ נסה לכתוב במחשב

1X = 2

(כדי לבדוק אם 1X יכול להיות שם של משתנה).
נסה כעת להדפיס את ערך המשתנה (PRINT 1X).

■ אמור למחשב להדפיס את התוכנית (LIST) וראה מה כתוב בשורה מספר 1 ... המחשב פשוט התייחס ל-1 כאל מספר שורה בתוכנית!

נסכם:

- שמו של משתנה חייב להתחיל באות אנגלית.
- אחרי אות זו יכולה לבוא אות נוספת או ספרה (כרצונך).

אגב: מותר לך לתת למשתנה שם ארוך יותר:

■ כתוב במחשב:

PGIOT = 3 (פגיעות = 3)

אך המחשב יתיחס רק לשתי האותיות הראשונות:

■ הדפס:

```
PRINT  PG
PRINT PGIOT
PRINT PG13
```

השם הכתוב על תא הזכרון של משתנה זה הוא ___ (PG, PGIOT)

אתגר מחשבתי

■ הגדר במחשב משתנה חדש, X, שיהיה שווה ל-10:

X = 10

(להזכירך: ניתן לכתוב X=10 במקום LET X=10 - וכמובן RETURN)

? מה יהיה ערכו של X לאחר שתפקוד על המחשב:

LET X = X+1

■ כדי לדעת אם צדקת, פקוד על המחשב לבצע פקודה זו (אפשר גם בלי LET) ואח"כ פקוד עליו להדפיס את X.

קיבלת כי X שווה ל _______ (1, 10, 11).

כיצד התקבלה תוצאה זו?

הפקודה X=X+1 אין פירושה ש-11=10!

הפקודה X=X+1 פירושה עבור המחשב הוראה לביצוע סדרה של פעולות:

- המחשב מבצע תחילה את מה שכתוב <u>בצד ימין</u> של השוויון: במקרה שלנו לוקח את הערך הנוכחי של המשתנה X ומוסיף 1: (10+1).
- את התוצאה המחשב <u>מציב</u> במשתנה ששמו רשום בצד שמאל של השוויון: במקרה שלנו מופיע שוב המשתנה X ולכן הוא מקבל עכשיו את הערך 11.

? מה תבצע התוכנית הבאה:

```
10 Y=1
20 PRINT Y
30 Y = Y+1
40 GO TO 20
```

לאחר שחשבת...

■ העתק את התוכנית למחשב והרץ אותה.

אגב: האם המספרים רצים מהר מדי על המסך?

■ החזק את מקש CTRL לחוץ כדי להאט את ריצתם למעלה.

נלך צעד־צעד ונראה כיצד התוכנית עובדת:

10 Y=1: אחרי ביצוע שורה זו המשתנה Y שווה ___ (3,2,1)

20 PRINT Y: המחשב מדפיס את ערכו של Y.

30 Y = Y+1: Y מקבל ערך חדש - ערך Y הקודם בתוספת 1:

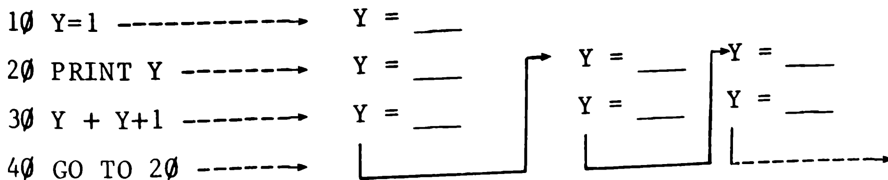
```
Y  =  Y + 1
↑     ↑
חדש   קודם
```

אחרי ביצוע שורה זו בפעם הראשונה

Y מקבל את הערך ___ (1, 2, 3)

40 GO TO 20: פקודה זו כבר מוכרת מיחידה 1. הפקודה שולחת את המחשב לשורה ___ בתוכנית.

■ עקוב אחר הגלגולים שעובר המשתנה Y בזמן ריצת התוכנית. היה <u>אתה</u> המחשב - עבור שורה שורה וכתוב את ערכו של המשתנה Y בכל שורה:



כדי לבדוק את תשובותיך ראה תשובה 8 בעמוד 57.

התוכנית עדיין רצה על המסך שלך?
אם לא תעצור את המחשב - האם הוא ימשיך לספור עד אינסוף?
כמעט! המחשב יכול לספור עד המספר העצום 10^{38}!
(הספרה 1 עם "זנב" של 38 אפסים)

רוצה הוכחה?

■ שנה את שורה 3Ø בתוכנית ל:

```
3Ø Y = Y*10
```

■ והרץ את התוכנית.

אתה רואה כי המספר הגדול ביותר שהמחשב מוכן לרשום הוא:
1E+38 $(1E+38 = 1\times10^{38})$

? מהו המספר הגדול ביותר שהמחשב רושם עדיין עם אפסים?
______________ (תשובה 9 בעמוד 57)

? חשוב: מה יקרה אם שורה 1Ø אינה קיימת?

משימות

■ בצע בתוכנית את השינויים הבאים:

- שנה את התוכנית כך שיודפסו רק המספרים האי-זוגיים.
- בצע שינוי כזה שיודפסו המספרים Ø, 5, 1Ø, 15 ...

? מה יקרה אם תשנה את שורה 4Ø:

```
4Ø GO TO 1Ø
```

פינת המשחק: "בול 500"

בעזרת מה שלמדת זה עתה תוכל לשחק משחק קטן עם עצמך או עם חבריך:

■ כתוב את התוכנית הבאה:

```
1Ø PRINT "{CLR/HOME}{2Ø →}{↓1Ø} BOOM!
2Ø B=1
3Ø B=B+1
4Ø GO TO 3Ø
```

■ הרץ את התוכנית.

? האם המחשב סופר על המסך? ________ (כן/לא)

דע לך כי מוחו של המחשב סופר ללא הרף במהירות רבה!

(המחשב מתרוצץ בין שורה 2Ø לבין שורה 3Ø)

רוצה להשתכנע שהוא סופר?

■ עצור את ריצתו ובקש ממנו להדפיס את ערכו של המשתנה B.

(אינך זוכר כיצד? ראה עמוד 10)

המספר שהודפס על המסך הוא ערכו של B כשהתוכנית בעצרה.

■ חזור והרץ את התוכנית. עצור את המחשב והצג שוב את ערכו של B. מה ערכו של B הפעם?

וכעת למשחק עצמו: "בול 500"

מטרת המשחק: לעצור את המחשב קרוב ככל האפשר למספר 500.

שיטת המשחק: כל אחד בתורו מקבל 5 נסיונות. בכל ניסיון הוא מריץ את התוכנית ועוצר אותה כאשר הוא חושב ש-B הגיע ל-500. כל אחד זוכר את השיא שלו. (המספר הקרוב ביותר ל-500).

מי שהגיע למספר הקרוב ביותר ל-500 - ניצח!

משתנה לולאה

מחק את התוכנית ועבור הלאה:

? מה תבצע התוכנית הזאת:

```
5   X = 1
1Ø  FOR N=1 TO 2Ø
2Ø  PRINT X
3Ø  X = X+1
4Ø  NEXT N
5Ø  PRINT "SOF"   (SOF = סוף)
```

לאחר שחשבת - העתק את התוכנית והרץ אותה.

למען האמת ניתן לקצר את התוכנית שכתבנו זה עתה ולכתוב אותה כך:

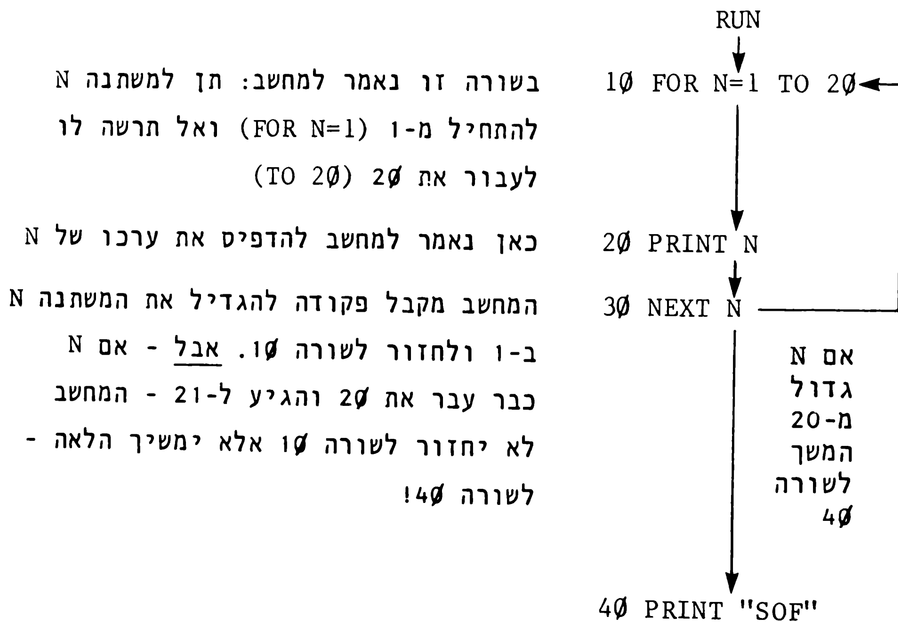
```
10 FOR N=1 TO 20
20 PRINT N
30 NEXT N
40 PRINT "SOF"
```

(תוכנית זו למעשה כבר הופיעה בתחילת הפרק, בעמוד 10).

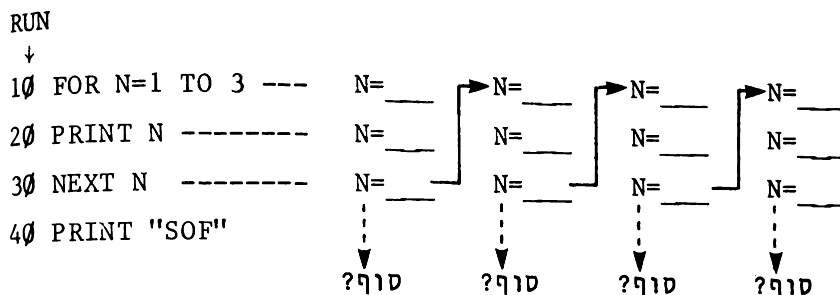
<u>שים לב לדבר מעניין:</u>

משתנה הלולאה N המופיע בשורה 20, מופיע גם בשורה 10 ובשורה 30!

כדי לעזור לך להבין את אשר קורה למשתנה N נעקוב אחר ביצוע התוכנית שורה אחר שורה:



? בדוק עצמך אם הבנת: הייה אתה המחשב וכתוב את ערכי המשתנה N במקומות הריקים:



החלט גם, מתי אתה כמחשב מדפיס "סוף".

(תשובה 10 בעמוד 57)

בחזור לתוכנית הנמצאת בזכרונו של המחשב:

? מה יקרה אם נשנה את שורה 10 ל:

10 FOR N=5 TO 20

התוכנית תספור מ ___ עד ___.

נסה וראה אם צדקת.

? מה יקרה אם תמחק לגמרי את שורה 30?

■ מחק את שורה 30 מן התוכנית והרץ.

המספר היחידי שהמחשב רשם הוא ___ (5, 10, 15, 20)

נוכחת לדעת כי ללא שורה 30 (NEXT) המחשב אינו חוזר כלל לשורה 10 (FOR ...) - הוא מדפיס את הערך הראשון בלבד של המשתנה N (במקרה זה - 5).

■ החזר את שורה 30 בחזרה לתוכנית.

יצירת השהייה

מה קורה כאשר אין כלל שורות בין צמד הפקודות FOR-NEXT היוצרות את הלולאה:

```
10 PRINT "START"      (START באנגלית - התחלה)
20 FOR T=1 TO 1000
30 NEXT T
40 PRINT "END"        (END באנגלית - סוף)
```

? מה עושה תוכנית זו? מה תפקידן של שורות 20 ו-30?
לאחר שחשבת...

■ נקה את הזכרון והכנס את התוכנית הזאת.

? מה עליך לעשות כדי להגדיל פי שניים את הזמן בין הופעת ה-START על המסך לבין ה-END?

(תשובה 11 בעמוד 57)

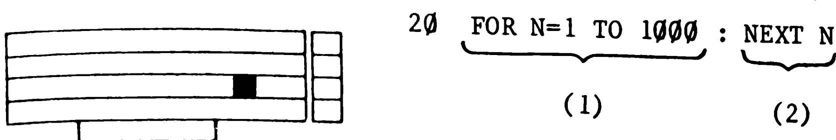
צמד השורות 20 ו-30 יצר השהייה.
כאשר המחשב מגיע לשורות 20 ו-30 הוא נמצא בלולאה, והוא מבצע אותה _____ (1000, 100, 10) פעמים. (המשתנה N מתחיל מ-1 ורק כאשר N גדול מ-1000 המחשב יוצא מן הלולאה.)

הזמן שבו ביצע המחשב 1000 סיבובים בלולאה יוצר השהייה.
מובן, לכן, שככל שהמחשב מבצע יותר סיבובים בלולאה - ההשהייה ארוכה יותר.

כתיבת מספר פקודות באותה שורה

עד כה נהגנו לכתוב בכל שורה בתוכנית פקודה אחת בלבד.
ניתן לכתוב מספר פקודות באותה שורה בעזרת שימוש בנקודותיים (:).

כך, למשל, ניתן לכתוב את שורות 2Ø ו-3Ø בשורה אחת:



```
2Ø FOR N=1 TO 1ØØØ : NEXT N
```

(1) FOR N=1 TO 1ØØØ

(2) NEXT N

כאשר המחשב גומר לבצע את קטע (1) בשורה הוא עובר לבצע את קטע (2), ולאחר מכן חוזר לקטע (1) וחוזר חלילה 1ØØØ פעמים!

רק כאשר N > 1ØØØ (N גדול מ-1ØØØ) המחשב ימשיך לשורה הבאה.

■ שנה את התוכנית כך, ששורה 3Ø תוכנס לתוך שורה 2Ø. הרץ את התוכנית וראה כי מתקבלת אותה תוצאה.

(האם זכרת למחוק את שורה 3Ø?)

■ כעת כתוב את כל התוכנית כולה בשורה אחת בלבד!

(תשובה 12 בעמוד 58)

שים לב:

כדאי להכניס מספר פקודות לאותה שורה רק במקומות שהדבר נוח (כמו פקודות להשגת פסק-זמן).

אל תרבה לרשום מספר פקודות בשורה - הדבר יקשה עליך לאתר "בגים".

ועוד הערה

יש גבול לאורך של שורה בתוכנית.

■ נסה לכתוב את השורה הבאה במחשב:

```
8Ø PRINT "{ 1ØØ → }"
```

? האורך המכסימלי של שורה בתוכנית הוא ___ (1, 2, 3) שורות על המסך.

התוכנית העייפה

התוכנית הבאה מסכמת את כל הפקודות והמושגים שלמדת עד כה ביחידת לימוד זו:

```
10 PRINT X ;
20 FOR T=1 TO X * X : NEXT T
30 X = X+1          ↑
                   כפול
40 GO TO 10
```

? מדוע קראנו לתוכנית זו ״התוכנית העייפה״?

הרץ את התוכנית ובדוק בעצמך.

(אינך מבין? פנה לתשובה 13 בעמוד 58)

ולסיום

מה למדת בפרק זה?

למדת שני מושגים חשובים מאוד בתיכנות:

1. הכרת את המשתנה (המספרי).
2. למדת להכיר לולאה סופית הנוצרת בעזרת צמד הפקודות FOR-NEXT. ראית כי שתי הפקודות חייבות לבוא האחת עם השניה.

בהמשך יחידת הלימוד נעשה שימוש נוסף במושגים שנלמדו.

פרק ג
בנית שעון דיגיטלי

מבוא

עכשיו, לאחר שהבנת את פקודות הלולאה FOR-NEXT, אתה יכול לבנות במחשב שלך שעון דיגיטלי.
אנו מתארים לעצמנו כי מצוי בביתך שעון דיגיטלי כלשהו.
רצוי מאוד ששעון כזה יהיה לידך בזמן שאתה בונה שעון דיגיטלי במחשב.

ספירת השניות

תחילה נתרכז בבניית הצגת השניות של השעון הדיגיטלי:

? התבונן, בשעון הדיגיטלי שלידך, על הספרות המציגות את השניות וענה:

השעון מתחיל לספור את השניות מ- ___ (Ø, 1).

האם מופיע גם המספר 6Ø בסוף הדקה? ___ (כן/לא).

אתה רואה כי השעון מתחיל מ-Ø ומגיע עד 59. ואז במקום לרשום 6Ø הוא חוזר ורושם Ø ומתחיל מן ההתחלה.
כדי לגרום למחשב להדפיס שניות על המסך נבנה את התוכנית בשלבים:

■ כתוב תוכנית שתדפיס, במרכז המסך בערך, מספרים המתחלפים מ-Ø עד 59.

זקוק לרמז? ראה בדף הבא.

רמז:

כדי שהמספרים יחליפו זה את זה, כלומר יודפסו זה על גבי זה עליך להביא את המסמן בתוכנית תמיד לאותו מקום על המסך, לפני שהוא מתחיל לכתוב את המספר.

■ הרץ את התוכנית שכתבת והתגבר על בגים, אם יש.
(לאחר מכן השווה לתשובה 14 בעמוד 58)

המספרים המופיעים על המסך מתחלפים במהירות אדירה.
■ הכנס השהייה לתוך התוכנית.

(תשובה 15 בעמוד 58)

■ הוסף לתוכנית שורה כזאת שתפקוד על המחשב לחזור ולספור מ-Ø בכל פעם שהוא מגיע ל-59.

(תשובה 16 בעמוד 58)

הכנסת השעון לקצב שניות אמיתי

? האם קצב הופעת המספרים על המסך ___ (מהיר יותר/איטי יותר) מקצב השניות בשעון הדיגיטלי שלידך?

? איזו השהייה עליך להכניס על-מנת להתאימו לקצב של שניות:

FOR T=1 TO _?_ : NEXT T

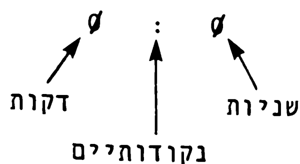
בצע כמה ניסויים עד שתגיע קרוב ככל האפשר לקצב שניות אמיתי.

(תשובה 17 בעמוד 58)

נפנה לספירת הדקות

התבונן בשעון הדיגיטלי שלידך: מה קורה כאשר השעון הדיגיטלי משלים לספור דקה שלמה?
במקרה זה סיפרת הדקות קופצת ב-1.

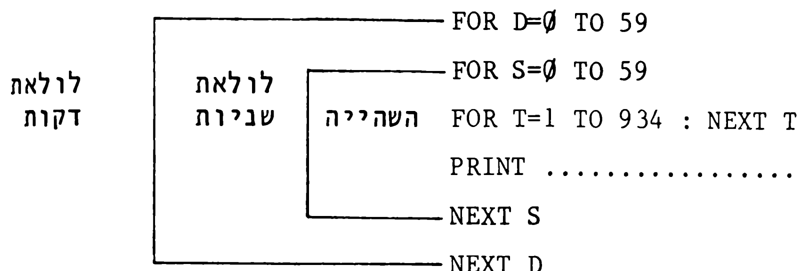
■ הוסף לתוכנית שלך שורות ושינויים כך, שהשעון יספור גם דקות:



(תשובה 18 בעמוד 58)

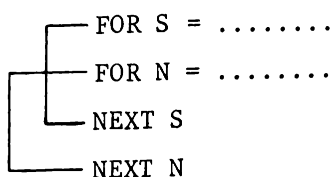
בוא נעצור לרגע ונתבונן בתוכנית שלנו.

בתוכנית יש 3 לולאות FOR-NEXT:



ניתן להשתמש בתוכנית במספר לולאות FOR-NEXT "השתולות" האחת בתוך חברתה.

? האם המחשב מסוגל לבצע תוכנית שבה לולאות מצטלבות:



(תשובה 19 בעמוד 59)

קיצור כתיבה של הקומודור

עד כה נהגנו לכתוב את שם המשתנה אחרי NEXT:

לדוגמא: NEXT S

למען האמת, הקומודור מוכן לקבל את פקודת ה-NEXT גם ללא שם המשתנה:

```
FOR T=1 TO 930 : NEXT
```

האם אין סכנה שהמחשב יתבלבל:

```
FOR D =
FOR S =
:
NEXT  →  NEXT של מה?
NEXT  →  NEXT של מה?
```

אל תדאג!

כבר אמרנו שאסור לבנות לולאות מצטלבות. ולכן, גם אם אינך כותב את שם המשתנה בפקודת NEXT, המחשב יודע למה התכוונת:

```
┌── FOR D =
│ ┌─ FOR S =
│ │   :
│ └─ NEXT
└── NEXT
```

המחשב משדך באופן אוטומטי את ה-FOR וה-NEXT הפנימיים ביותר וכך הלאה...

המלצה: אל תתעצל וכתוב את שם המשתנה בפקודת NEXT במקומות בהם אתה עלול להתבלבל.

■ הוסף לשעון גם שעות!

הפתעה!

■ נקה את הזכרון, העתק את התוכנית הבאה והרץ אותה:

```
10 PRINT T I $ ; "{HOME}" : GO TO 10
```

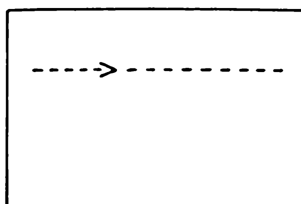
? מה קורה כאן?

ראה תשובה 20 בעמוד 59

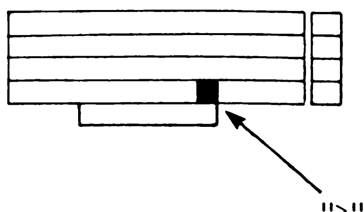
פרק ד
אנימציה – יצירת תנועה על המסך

זוכר את אי-ציק שעשה התעמלות בוקר ביחידה 1?
עכשיו תנצל את הכלים החדשים שרכשת על-מנת להניע בצורה מבוקרת צורות על פני כל המסך.

משימה 1



עליך לכתוב תוכנית אשר תניע את הצורה ">" לרוחב המסך משמאל לימין. כאשר הצורה ">" תגיע לקצה הימני של המסך היא תחזור ותנוע שוב משמאל לימין - אבל באותו גובה ולא תרד שורה!



דע לך כי אתה כבר יודע את כל הפקודות הדרושות לביצוע המשימה. נסה לכתוב את התוכנית לבדך:

10 ____________________

20 ____________________

⋮ ____________________

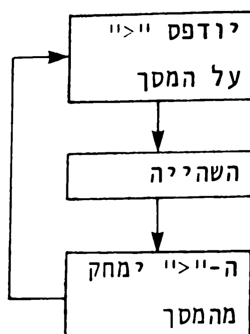
____

____

צבע את הציפור כרצונך.

אם אתה מתקשה - יש רמזים
בעמוד הבא...

רמז: כדי ליצור אפקט של תנועה על המסך, עליך לבנות את התוכנית כך ש:

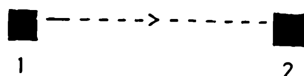


- כמה פעמים צריך לחזור על פעולות אלו על-מנת לקבל תנועה לרוחב כל המסך?
- מה צריך להוסיף כדי שהציפור תעוף שוב ושוב משמאל לימין?
- האם הציפור יורדת בכל זאת שורה בכל פעם שהיא חולפת על פני המסך? תקן זאת!

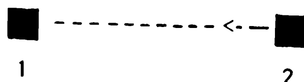
(תשובה 21 בעמוד 59)

משימה 2

עליך לשנות את התוכנית כך שהמחשב יצייר 2 "קירות" על המסך:



הציפור יוצאת מקיר 1 מגיעה לקיר 2, "מסתובבת" נהפכת לצורה (<), וחוזרת:



מגיעה לקיר 1, שוב "מתהפכת" וכך הלאה...

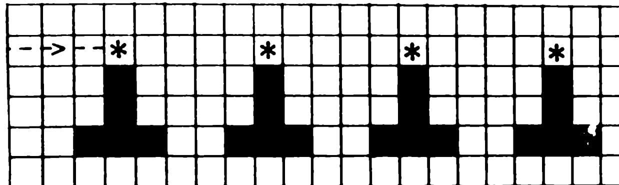
בבנית התוכנית הקפד:

- ״הציפור״ אינה מוחקת את ״הקירות״.
- ״הציפור״ עפה הלוך ושוב ללא הפסק.
- הקירות אינם מהבהבים.
- הציפור מגיעה במעופה ממש עד הקירות.
- צבע את הקירות ואת הציפור כרצונך.
- (אולי תהיה מעונין לשנות את צבע הציפור בכל כיוון?)

(תשובה 22 בעמוד 59)

משימה 3

כתוב תוכנית שתצייר 4 נרות:



לאחר ציור הנרות, מופיעה רוח (>), עפה ימינה, ומכבה את הנרות.

הנה כמה עצות:

- הרוח נעה כמו הציפור בתוכנית הקודמת.
- כתוב תוכנית שמשכפלת את 4 הנרות כמו ששיכפלת את אי-לי.
- צבע את התוכנית כרצונך.

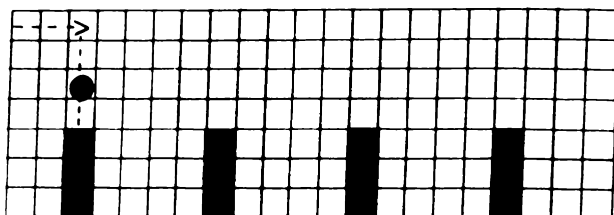
(תשובה 23 בעמוד 60)

משימות נוספות עם תוכנית הנרות:

- במקום הרוח - שמך נע ימינה ומכבה את הנרות.
- לאחר כיבוי כל הנרות - התוכנית באופן אוטומטי חוזרת על עצמה - הנרות נדלקים באופן אוטומטי והרוח מכבה אותם...
- התוכנית מתחילה עם נר אחד בלבד - לאחר כיבוי הנר התוכנית מציירת 2 נרות והרוח (או שמך) מכבים אותם, וכך הלאה עד 5 נרות - ואז התוכנית נעצרת.

עמודים נעלמים...

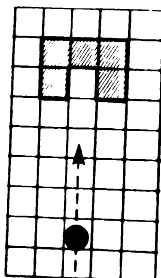
- על המסך מופיעים 4 עמודים:



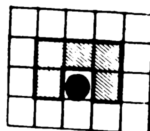
הציפור מופיעה ועפה ימינה, נעצרת מעל לעמוד הראשון, מפילה כדור הנופל למטה ומוחק את העמוד. אחרי שהעמוד נמחק כולו, הציפור ממשיכה במעופה לעמוד השני, נעצרת בדיוק מעליו, מפילה כדור המוחק גם עמוד זה וכך הלאה!

פינת המשחק: חור ברשת...

התוכנית מציירת על המסך שער, וכדור מתחיל לעלות לעברו כלפי מעלה:



מטרת המשחק: לעצור את הכדור (בעזרת מקש: RUN/STOP*) בדיוק בתוך השער:*



לפני שהכדור יחלוף ו"יקרע" את הרשת...

<u>הערה</u>: בהמשך תלמד איך לעצור את הכדור - בלי לעצור את התוכנית!

פרק ה
ריבועי קסם

זוכר שעסקת בשכפול צורות בפרק א' בעזרת FOR-NEXT?
נתחיל את הפרק בכמה משימות שכפול:

■ כתוב תוכנית שתשכפל $ אחד כך שתקבל שורה של דולרים (בעזרת FOR-NEXT):

בתוכנית לא צריך להופיע יותר מ-$ אחד.

(תשובה 24 בעמוד 60)

■ כעת הוסף לתוכנית עוד לולאת FOR-NEXT כך, שיתקבל שטיח (ריבוע) של דולרים:

זכור:

- בתוכנית מותר רק ל-$ אחד להופיע!
- עליך לחשוב כל הזמן איפה נמצא המסמן.

לאחר שניסית והתמודדת עם הבגים - השווה לתשובה 25 בעמוד 60

■ הכנס בתוכנית שינויים כאלה, שהשטיח יהיה בגודל 5×5.

■ שנה את התוכנית כך, שיודפס שטיח של 2∅ שורות ו-2∅ עמודות.
קרוב לודאי שאתה נאלץ לבצע שינויים בשני מקומות בתוכנית, בכל פעם שאתה רוצה לשנות את גודל הריבוע:

■ שנה את התוכנית כך, שכדי לשנות את גודל השטיח תצטרך לשנות מספר רק במקום אחד בתוכנית!

(תשובה 26 בעמוד 60)

משכללים את התוכנית

אנו מעוניינים שהמחשב ישאל אותך לפני ציור השטיח באיזה גודל שטיח אתה מעוניין.

לשם כך עליך ללמוד פקודה חדשה:

INPUT (אִינְפּוּט) – הַכנס לתוכנית:

■ שנה את שורה 5:

5 INPUT L (ראה תשובה 26 בעמוד 60)

שים לב: אין קיצורי-כתיבה לפקודה INPUT - כתוב אותה במלואה.

■ הרץ את התוכנית.

הופיע סימן-שאלה והמסמן מהבהב:

המחשב שואל אותך איזה ערך אתה רוצה שיהיה למשתנה L :(INPUT L)

■ הקש את המספר (גודל הריבוע) ואל תשכח RETURN.

המחשב שרטט ריבוע בדיוק במידה שבחרת!

כיצד קרה הדבר?

כאשר המחשב הגיע לשורה: 5 INPUT L הוא נעצר, הציג על המסך סימן-שאלה, וחיכה שתכניס לו את הערך של המשתנה L.

לאחר שהכנסת מספר, הוא שומר בזכרונו כי המשתנה L שווה למספר זה והוא ממשיך בביצוע התוכנית.

בכל מקום בתוכנית בו מופיע המשתנה L, הוא שווה למספר שהכנסת ב-INPUT.

■ שנה את התוכנית כך, שבסיום של הדפסת כל ריבוע המחשב ישאל אותך בפינה השמאלית-עליונה מה הגודל בו אתה רוצה את הריבוע הבא. כל זאת - בלי למחוק את הריבוע שכבר מצויר. כמובן, שלאחר שהכנסת מספר, הריבוע הקודם נמחק - ומודפס ריבוע חדש בגודל הרצוי (ושוב מופיע סימן השאלה של ה-INPUT בפינה השמאלית-עליונה).

(תשובה 27 בעמוד 61)

עצירת התוכנית במצב INPUT

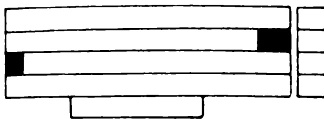
■ הרץ את התוכנית. המחשב שואל אותך, כמובן, מה גודל הריבוע הרצוי לך.

במצב זה - נסה לעצור את התוכנית ולהציג אותה (LIST).

התוכל?

קשה... נסה את השיטה הבאה:

■ לחץ RUN-STOP וגם RESTORE בעת ובעונה אחת:



שכלול מדהים!

רוצה שהמחשב גם ישאל אותך מאיזה סימן אתה רוצה שהוא יבנה את הריבוע? (במקום $)

■ הוסף לתוכנית עוד פקודת INPUT:

```
7  INPUT A $
```

(אל תשכח לכתוב $)

שנה גם את שורה 3Ø ל:

```
3Ø PRINT A$ ;
```

■ לאחר שהכנסת את השינויים הרץ את התוכנית.

בתחילה המחשב שואל אותך מה ערכו של המשתנה L - גודל הריבוע. בחר במספר ולחץ RETURN -

עכשיו מופיע עוד סימן-שאלה - המחשב רוצה לדעת מאיזה סימן לבנות את הריבוע. בחר סימן כלשהו ולחץ שוב RETURN - המחשב הממושמע משרטט ריבוע בגודל ובצורה שפקדת עליו!
לפני שנכנס להסבר מפורט - שחק להנאתך עם התוכנית:

- הכנס צורות שונות.
- הכנס גדלי-ריבוע שונים.
- בדוק מה קורה אם מכניסים 2 צורות ביחד.
- שנה את צבע המסמן וראה מה קורה.
- אל תכניס דבר - לחץ פעמיים RETURN!

נהנית? נעבור להסבר יותר מפורט:

משתנה מחרוזת

בשורה 5 יש משתנה L שקלט את המספר שהכנסת:

```
5  INPUT L
```

בשורה 7 (ובשורה 3Ø), לעומת זאת, מופיע משתנה מסוג חדש הנקרא משתנה מחרוזת:

```
7  INPUT A$
```

מהי מחרוזת?

למען האמת, כבר השתמשת בעבר במחרוזות, מבלי לדעת שהן "מחרוזות":

? מה מבצע המחשב בפקודה הבאה:

```
PRINT "2+5"
```

המחשב אינו מבצע את פעולת החשבון אלא פשוט מעתיק את כל מה שכתוב בתוך המרכאות.
כל דבר שתשים בתוך מרכאות יחשב כמחרוזת:

"מחרוזת"

? בפקודה הבאה:

PRINT "4*7=" ; 4*7

איזה חלק הוא מחרוזת ואיזה חלק הוא ביטוי חשבוני?

(תשובה 28 בעמוד 61)

ומהו משתנה מחרוזת?

כתוב במחשב:

LET B$ = "XØR" (אל תמחק את התוכנית שישנה בזכרון)

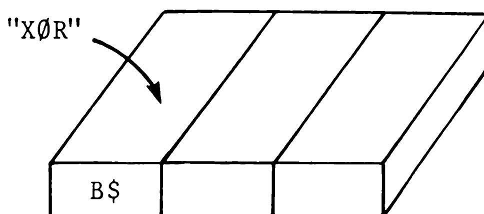
? מה יקרה אם תפקוד:

PRINT B$

נסה. כיצד תסביר את התוצאה?

התהליך דומה מאוד למה שקורה במחשב עם משתנים מספריים:

המחשב קרא לאחד התאים שבזכרונו: B$ והכניס לתוכו את המחרוזת "XØR"



כאשר פקדת PRINT B$ המחשב מצא את התא הנקרא B$, העתיק מתוכו את המחרוזת והדפיס אותה על המסך.

? מה יקרה, לדעתך, אם תכתוב:

LET B$ = 5

נסה ותיווכח אם צדקת.

המחשב הודיע לך TYPE MISMATCH - כלומר: אין התאמה בין השם של התא בזכרון (B$ - משתנה מחרוזתי) לבין מה שניסית להכניס לתוכו (5 - ערך מספרי).

? האם המחשב יקבל "5" = LET B$? נסה!

נסה להכניס מחרוזת לתא של משתנה מספרי (למשל "5"=LET B).

? מה תעשה תוכנית שכזאת:

```
500  X$ = "שמך"
510  FOR N=1 TO 10
520  PRINT X$
530  NEXT
```

כדי לבדוק אם אתה צודק אל תמחק את התוכנית הנמצאת כבר בזכרון!

■ העתק את התוכנית, כתוב RUN 500 ולחץ RETURN.

המחשב יתחיל "לרוץ" משורה 500 ואילך, וכך תוכל לבדוק אם צדקת לגבי התוכנית הקטנה הכתובה למעלה.

■ סיימת? מחק את השורות מ-500 ואילך.

שמות של משתני מחרוזת

הכללים החלים על שמות של משתנים מספריים נכונים גם לגבי משתני-מחרוזת:

- השם חייב להתחיל באות אנגלית.
- אחרי אות זו יכולות לבוא אותיות וספרות כרצונך.
- המחשב מזהה רק את שני הסימנים הראשונים של השם ומתעלם מן השאר.

אבל:

שם של משתנה מחרוזת חייב להסתיים תמיד בסימן $!

? מתוך השמות הבאים של משתני מחרוזת - איזה חוקיים ואיזה המחשב לא יקבל:

A$, 2A$, AAA$, B3, C$$, C*$

(תשובה 29 בעמוד 61)

המחרוזת הריקה

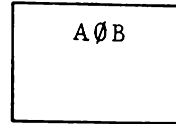
זוכר מה קרה כאשר ביקשת מהמחשב להדפיס משתנה מספרי שלא הגדרת קודם לכן מה ערכו?

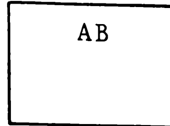
הוא פשוט הדפיס Ø.

ואיך יגיב המחשב על משתנה מחרוזת שלא הוגדר?

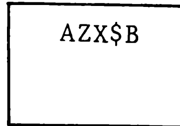
? נחש מה יופיע על המסך כאשר תפקוד על המחשב:

PRINT "A" ; ZX$; "B"

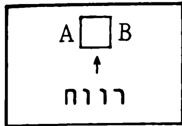




ב



ג



ד

לאחר שניסית לנחש...

כתוב PRINT "A" ; ZX$; "B" (ולחץ RETURN).

הסבר

ZX$ הוא משתנה-מחרוזתי שלא הוגדר עדיין (לא קיבל כל ערך קודם). במקרה כזה - כאשר אתה לא נותן למשתנה מחרוזתי ערך כלשהו - הוא מקבל את הערך של המחרוזת הריקה:

ZX$ = ""

מחרוזת ריקה היא מחרוזת שאינה מכילה כלום בין המרכאות - אפילו לא רווח (ובודאי לא Ø).

וחזרה לתוכנית שלנו:

בתוכנית שלנו רצינו שהמחשב ישרטט את הריבוע עם אחד מהסימנים המופיעים על המקשים - אות, מספר, צורה גרפית ועוד.

לשם כך השתמשנו במשתנה מחרוזת A$.

כאשר המחשב הגיע לשורה:

```
7  INPUT A$
```

הוא מציג סימן-שאלה על המסך ומחכה שתגיד לו למה אתה רוצה שמשתנה המחרוזת A$ יהיה שווה.

שים לב: אינך צריך, במקרה זה, לרשום מרכאות - המחשב יודע כי עליו לקלוט מחרוזת.(איך הוא יודע?)

לאחר שהקשת על אחד הסימנים (עם RETURN, כמובן), המחשב יודע כי משתנה המחרוזת A$ שווה לסימן שהקשת.
בכל פעם שהוא מגיע לשורה

```
30 PRINT A$ ;
```

הוא מדפיס את הסימן שתקתקת ל-INPUT A$.

? מה יקרה אם משתנה המחרוזת A$ יהיה שווה למחרוזת בת 2 סימנים? (למשל: "A$ = "AB)

נסה!

כותרות ל־INPUT

לתוכנית שלך יש עכשיו שתי שורות עם פקודות INPUT:

- עבור המשתנה המספרי L.
- עבור המשתנה המחרוזתי A$.

שמת בודאי לב, כי סימני-השאלה שהמחשב מציג הם זהים. כדי לא להתבלבל בין שני ה-INPUT אפשר להורות למחשב להדפיס כותרת מיוחדת לכל סימן-שאלה שהוא מציג, וכך נדע בדיוק לאיזה נתון הוא מחכה.

הנה דוגמא:

שנה את שורה 5 :

המשתנה נשאר L ← `5 INPUT "L=" ; L`

הכותרת היא מחרוזת

אל תשכח

■ הרץ את התוכנית.

עכשיו מתקבל: L = ?

ואתה יודע בדיוק לאיזה משתנה המחשב מחכה.

■ עצור את ריצת התוכנית. (אינך זוכר? ראה עמוד 34)

עכשיו הכנס כותרת גם ל-INPUT השני.

לדוגמא:

```
7  INPUT "SIMAN=" ; A$
```

סימני פעולה בתוך מחרוזת

אתה כבר יודע כי כאשר המחשב מגיע למחרוזת הוא מעתיק את הסימנים בתוך המחרוזת כמו תוכי.

יש יוצאים מן הכלל וגם בהם נתקלת.

כתוב במחשב:

```
PRINT "123{↓3}456"
```

על המסך אתה רואה:

```
PRINT "123QQQ456"
```

? האם תוכל לשער מה יודפס על המסך לאחר שתלחץ RETURN?

■ לחץ RETURN וראה אם צדקת.

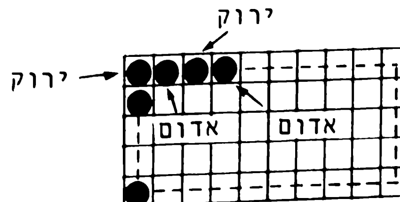
הסבר:

כאשר המחשב הגיע לשלושת סימני הפעולה (QQQ) שבתוך המחרוזת הוא לא העתיק אותם אלא ביצע אותם והוריד את המסמן 3 שורות למטה.

דבר זה נכון גם לגבי הוראות צבע הנמצאות בתוך מחרוזת:

משימה

כיצד תוכל, בעזרת התוכנית הנמצאת כבר במחשב, להדפיס מלבן כזה:

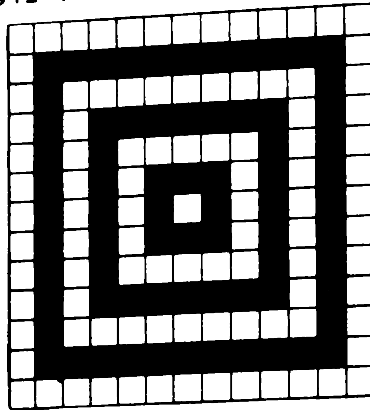


(תשובה 30 בעמוד 61)

ריבועים מתנפחים

נקה את זכרון המחשב מן התוכנית הקודמת.

עליך לכתוב עכשיו תוכנית שי"תנפחי" ריבועים על המסך:



התוכנית מציירת תחילה את הריבוע הפנימי, אחר כך את הריבוע הבא וכן הלאה.

יש בידך את כל הכלים בשפת ה-BASIC הדרושים לבנית התוכנית.

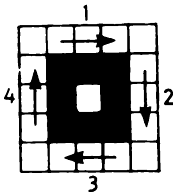
גש לעבודה.

אתה מתקשה?

תוכל להעזר ברמזים הבאים:

רמז 1

בנה תחילה את הריבוע הפנימי בעזרת 4 לולאות FOR-NEXT נפרדות:



הערה: דאג לצייר את הריבוע במרכז המסך.

התוכל להמשיך לבדך?

רמז 2

הוסף לתוכנית שינויים כאלה, שהמחשב ישאל אותך איזה גודל ריבוע אתה רוצה - ואח"כ יצייר אותו בגודל הרצוי!

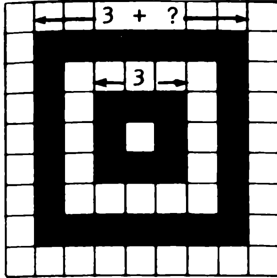
הערה: בדוק אם המחשב אכן מצייר את הריבוע <u>בדיוק</u> בגודל הרצוי...

(תשובה 31 בעמוד 61)

רמז 3

מכאן הדרך למטרה די פשוטה:

במקום פקודת ה-INPUT עליך לדאוג לכך שהמשתנה L יגדל לאחר ציור כל ריבוע ב:



כמו כן עליך לדאוג, שבגמר ציור כל ריבוע, המסמן ינוע לתחילת הריבוע הבא!

(תשובה 32 בעמוד 62)

מסגרות לפי מידה

עליך לפתח תוכנית שתבצע את הפעולות הבאות:

- המחשב שואל אותך איזה שם אתה רוצה להכניס למסגרת.
- אתה כותב את השם - ואז המחשב כותב אותו במרכז המסך ומצייר מסביבו מסגרת המתאימה את עצמה באופן אוטומטי לגודל השם שבחרת!

לדוגמא:

עבור שם קצר - מסגרת קצרה:

```
* * * * * * * *
*             *
*   D A N I   *
*             *
* * * * * * * *
```

ועבור שם ארוך יותר - מסגרת ארוכה:

```
* * * * * * * * * * *
*                   *
*   K A L I S K Y   *
*                   *
* * * * * * * * * * *
```

בנוסף לכל הידע שצברת עד כה, עליך להכיר את הפקודה המחשבת את אורך המחרוזת, כלומר את מספר הסימנים המופיעים במחרוזת:

LEN (" ....")

מחרוזת

LEN הוא קיצור של המילה LENGTH - **אורך**.

כדי להבין את פעולת הפקודה LEN -

כתוב במחשב:

```
PRINT LEN ("ABCDE")
```

? לאחר שלחצת RETURN המחשב כתב את המספר ______.

מספר זה הוא בדיוק אורך המחרוזת (מספר הסימנים במחרוזת).

? מה יכתוב המחשב על המסך לאחר ביצוע התוכנית הבאה:

```
10 B$ = "A123CH5"
20 PRINT LEN (B$)
```

ועכשיו - לאחר שהכרת את פקודת LEN - אתה יכול להשתמש בה על מנת לבנות את תוכנית-המסגרות.

אנו מציעים כי תעיז ותנסה להתמודד עם התוכנית לבדך - אפילו אם הדבר יגזול ממך זמן מה.

אם בכל זאת אתה מתקשה - פנה לרמזים הבאים:

- עיקר הקושי בתוכנית הוא להתאים את אורך המסגרת לאורך השם:

  הנה דוגמא:

  אורך השם הוא 4

  אורך המסגרת הוא ___ (8,4)

```
* * * * * * * *
*             *
*   D A N I   *
*             *
* * * * * * * *
```

אורך המסגרת

- אורך המסגרת = אורך השם + ____ (4, 8, 2)

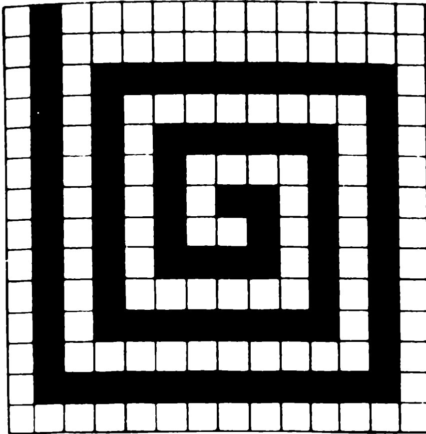
  (עדיין מתקשה? ראה תשובה 33 בעמוד 62)

שכלולים

- המחשב ישאל אותך באיזה צבע אתה רוצה את השם.
- כנ"ל לגבי צבע המסגרת.
- המחשב ישאל אותך מאיזה סימן לבנות את המסגרת!

שבלולים מתנפחים

כתוב תוכנית שתצייר על המסך שבלול:

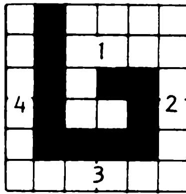


כל הכלים בידך.

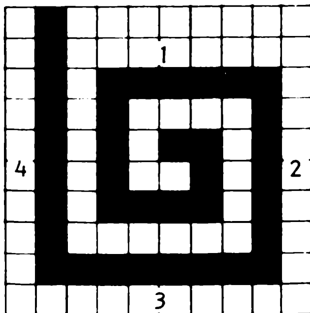
צא לדרך!

אתה זקוק בכל זאת לרמז?

דע לך, כי בדומה לתוכנית של הריבועים המתנפחים, גם התוכנית הזאת בנויה מצורה יסודית אחת:



ההולכת ומתנפחת:



כתוב תוכנית, כללית, המציירת את הצורה היסודית, ותן לה פקודה "להתנפח" שוב ושוב...

(תשובה 34 בעמוד 63)

פרק ו
עפרון וידיאו

המשימה המרכזית בפרק זה תהיה לפתח תוכנית "עיפרון-וידיאו" שתבצע את הפעולה הבאה:

1 הרצת את התוכנית: הופיע הסימן "X" במרכז המסך.

RUN: X

2 לחצת על המקש המזיז את המסמן ימינה: טור של סימני "X" נע מיד ימינה

CRSR ⇒ : XXXX

3 לחצת על המקש המזיז את המסמן למעלה: טור ה-"X" נע למעלה

CRSR ⇑ :

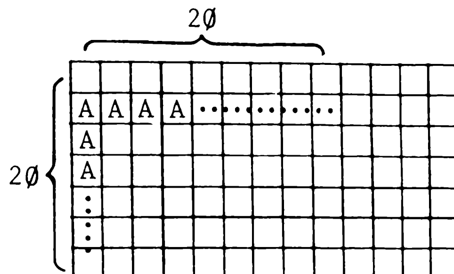
```
      X
      X
      X
X X X X
```

וכך הלאה לגבי כל מקשי-המסמן.
בעזרת תוכנית זו תוכל לצייר כאוות נפשך על המסך בקלות ובמהירות.

עדיין אינך יכול לבנות את תוכנית עיפרון-הוידיאו.
חסרה לך עדיין פקודה שלא למדת.

תרגיל הכנה:

■ בנה תוכנית שתצייר ריבוע של 2Ø×2Ø משבצות:



התוכנית שואלת אותך בכל פעם מאיזה סימן לבנות את הריבוע הבא (INPUT). אחרי שאתה עונה המחשב מדפיס ריבוע חדש באותו גודל - אבל בנוי מהסימן שהכנסת ב-INPUT - וכך הלאה. זו לא תוכנית חדשה בשבילך...

... השווה אותה לתשובה 35 בעמוד 63

GET A$

כעת אנחנו מגיעים לפקודה החדשה שבשבילה בנית את כל התוכנית:

■ במקום השורה INPUT A$ שבתוכנית שלך
כתוב: GET A$

אגב: אם למשתנה שלך קראת בשם אחר, למשל INPUT C$, כמובן שתשנה את השורה ל-GET C$.

■ הרץ את התוכנית, לחץ על מקשים כרצונך.
אתה רואה כי מתקבלת אותה תוצאה - בהבדל אחד:

- אינך צריך ללחוץ RETURN בכל פעם!

■ לחץ במהירות, בזה אחר זה, על הספרות מ-1 עד 9, עקוב אחר ההתרחשויות על המסך.

אתה רואה כי המחשב "זוכר" את הלחיצות שלך בדיוק בסדר שבו תיקתקת!

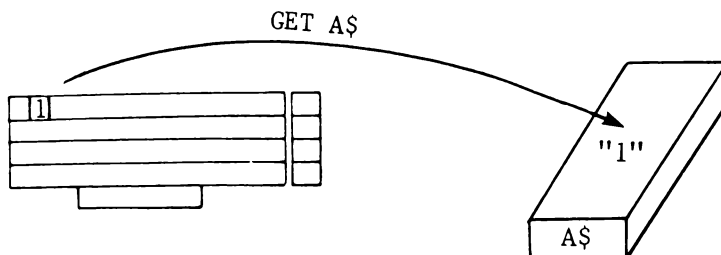
הבה נראה כיצד פועלת פקודת GET A$.

(אגב - במקום A$ אפשר לכתוב B$ או כל שם חוקי אחר של משתנה):

הסבר

כשהמחשב מגיע בתוכנית לפקודה GET A$ הוא מכניס לתא של המשתנה A$ את הסימן הראשון עליו הקשת:

לדוגמא:



עכשיו A$ שווה ל ______ ("", "2", "1").

כעת המחשב ממשיך בביצוע התוכנית ומצייר ריבוע הבנוי מהסימן הנמצא בתא של המשתנה A$. כאשר הוא חוזר שוב לפקודת GET A$ הוא לוקח את הסימן הבא שהקשת ומכניס אותו לתא A$. נניח שהסימן השני שהקשת היה "2". כעת A$ שווה ל ____ ("", "2", "1").

וכן הלאה עד "שניגמרים לו" הסימנים שהקשת.

אז קורה דבר מעניין:

המחשב לוקח את המחרוזת הריקה ("") ומכניס אותה לתא A$.

כעת A$ שווה ל ___ ("", "2", "1").

המחשב ממשיך בתוכנית, וכשהוא מגיע לפקודה:

PRINT A$

הוא אינו מדפיס כלום! לכן גם הריבוע הקודם אינו נמחק. כך הוא ממשיך ומתרוצץ "על-ריק" בתוכנית עד שאתה מקיש שוב על אחד המקשים.

? מה יקרה לדעתך אם תלחץ על מקש הרווח? נסה!

נסכם:

GET דומה מאוד ל-INPUT, בכמה הבדלים:

- GET אינו מחכה לך! המחשב חולף במהירות על פני פקודת ה-GET בתוכנית. אם לחצת קודם על מקש כלשהו - מקש זה נכנס לתא של משתנה ה-GET. אם לא לחצת - המחשב משתמש במחרוזת הריקה.
- ה-GET מכניס לתא של המשתנה רק <u>סימן אחד</u> בכל פעם: בניגוד ל-INPUT, הרי שב-GET לא תוכל להכניס לתא של המשתנה את שמך, למשל.

כתיבת התוכנית

כעת תוכל לגשת לכתיבת התוכנית של עיפרון-הוידיאו.

(אינך זוכר את המשימה? חזור ועיין בעמוד 45)

זקוק לרמזים? הנה הם לפניך:

- אחרי ביצוע הפקודה ; PRINT "X{$\overset{1}{\leftarrow}$}", איפה יהיה המסמן יחסית ל-X? __________ (מימין ל-X, על ה-X, משמאל ל-X).
- אחרי ביצוע הפקודה ; PRINT "X{$\overset{1}{\leftarrow}$}";"{↑1}", איפה יהיה המסמן יחסית ל- ? _____ (מעל ל-X, על ה-X, משמאל ל-X).
- אם A$ = "{↓1}", איפה יהיה המסמן לאחר ביצוע:

  PRINT "X{$\overset{1}{\leftarrow}$}" ; A$;

  ________ (מעל ל-X, מתחת ל-X, על ה-X).

אתה רואה ש-A$ קובע את הכיוון אליו ינוע המסמן לאחר הדפסת ה-X.

? האם תוכל כעת לכתוב את תוכנית עיפרון הוידיאו?

רמז אחרון

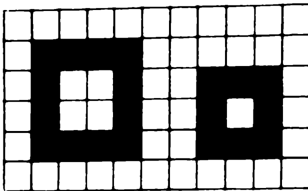
השלם את התוכנית הזאת - ותקבל עיפרון וידיאו:

```
10 GET ________
20 PRINT "X{←}" ; ______ ;
30 GO TO ________
```

(תשובה 36 בעמוד 64)

? האם תוכל לשנות את צבע "העיפרון" ולהדפיס ציור המורכב מכמה צבעים?

? איך אפשר, בעזרת עיפרון-הוידיאו שבנית, לצייר את הציור הבא:



(שני מלבנים לא מחוברים)

משימה

עליך לבצע בתוכנית שינוי כזה, שיאפשר להחליף את "חוד העיפרון", כלומר את ה-X איתו אתה מצייר - לכל סימן שתבחר מלוח המקשים - וכל זאת בלי לעצור את התוכנית ובלי למחוק את הציור שכבר קיים על המסך!

לשם כך עליך ללמוד פקודה חדשה:

IF... THEN... (אם... אז...)

? מה תבצע התוכנית הבאה:

```
10 GET C$
20 IF C$ = "N" THEN PRINT "שמך"
30 GO TO 10
```

שורה 20 פירושה: אם (IF) "C$="N אז (THEN) הדפס את שמך.

? נסה תחילה לחשוב מה תבצע התוכנית על המסך ואז...

■ העתק את התוכנית למחשב והרץ אותה.

לחץ על מקשים שונים. לחץ על המקש "N" מה קורה?

הסבר

בכל פעם שהמחשב מגיע לשורה 2Ø הוא בודק אם (IF):

C$ = "N"

- אם לא - כלומר לא הקשת כלל או הקשת על מקש אחר מאשר "N" - המחשב עובר הלאה לשורה 3Ø ואינו מדפיס דבר.
- אבל - אם התנאי מתקיים, כלומר, הקשת "N" ולכן C$ = "N" - רק אז המחשב ממשיך בשורה 2Ø, ומבצע:

PRINT "שמך"

אתה יודע כל מה שצריך לדעת על-מנת להחליף את חוד העיפרון מ-X לסימן אחר.

אבל כיון שפקודת ה-IF ... THEN כל כך חשובה בשפת התיכנות, הבה נתרגל אותה עוד קצת.

? מה תבצע התוכנית הבאה:

```
10 GET D$
20 IF D$ = "" THEN GO TO 10
30 PRINT "***"
40 GO TO 10
```

(הדפס כאן מחרוזת ריקה)

לאחר שיש בידך השערה מבוססת:

■ הרץ את התוכנית. לחץ על מקשים שונים ובדוק אם צדקת בהשערתך.

משימה

כתוב תוכנית שתדפיס את המספרים מ-1 עד 5Ø על המסך, אחד מתחת לשני - ותיעצר.

אבל - אסור לך להשתמש בהוראות FOR-NEXT!

הערה: הפקודה END - עוצרת את התוכנית!

(תשובה 37 בעמוד 64)

משימות נוספות

■ עליך לכתוב תוכנית שתבצע את כל הפעולות הבאות:

- בכל פעם שתלחץ על המקש של האות הראשונה של שמך, התוכנית תדפיס את שמך המלא.
- כנ"ל לגבי שמות נוספים שתבחר.
- בכל פעם שתלחץ על המקש "L", המחשב ידפיס את התוכנית עצמה על המסך (ויעצר).
- התוכנית תכיל מנגנון השמדה-עצמית: אם תלחץ על המקש "N" התוכנית תשמיד את עצמה! (NEW).

(תשובה 38 בעמוד 64)

פינת המשחק:

■ כתוב תוכנית אשר תאפשר לך לשחק את המשחק הבא נגד חברך:

- חברך יוצא מן החדר, ואתה מריץ את התוכנית ומכניס, ב-INPUT, מספר "סודי" (עד 99) לתוך המחשב. התוכנית מוחקת את המספר מן המסך.
- חברך חוזר אל החדר, מתישב מול המחשב. המחשב שואל אותו מה הניחוש שלו לגבי המספר הסודי שהכנסת.
- אם הניחוש גבוה מן המספר הסודי אז המחשב מודיע: "גבוה מדי".

  אם הניחוש נמוך מן המספר הסודי אז המחשב מודיע: "נמוך מדי".

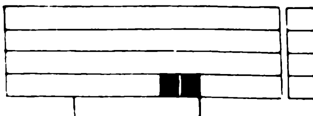
  אם הניחוש מדויק אז המחשב מודיע לו שקלע בול!
- אח"כ אתם מתחלפים ועליך לנחש את מספרו הסודי של חברך.

מטרת המשחק - לנחש את המספר הסודי במינימום ניחושים!

רמז

IF A>B פירושו: אם A גדול מ-B

IF A<B פירושו: אם A קטן מ-B



חוזרים לעיפרון הוידאו

כעת, לאחר שלמדת את פקודת IF-THEN, אפשר לחזור למשימה המקורית: החלפת חוד-העיפרון (ראה עמוד 49).

■ בצע כעת את המשימה שבעמוד 49.

זקוק לרמזים?

רמז 1

```
5 ..............
10 GET A$
15 אם A$ = "C" אז לך לשורה 100
20 PRINT .......
30 GO TO 10

100 ............
:
150 GO TO 10
```

תוכנית עיפרון-הוידיאו (שורות 5–30)

החלפת חוד-העיפרון (שורות 100–150)

רמז 2

בשורה 20, במקום "X" (חוד העיפרון) עליך לכתוב משתנה מחרוזת.

רמז 3

בשורה 5 עליך לתת לחוד העיפרון ערך ראשוני כלשהו.

(תשובה 39 בעמוד 64)

מחשבה נוספת על פקודת IF... THEN

? מה תבצע תוכנית זו:

```
10 GET A$
20 IF A$ = "N" THEN PRINT "***" : GO TO 10
```

אל תעבור הלאה בלי לבדוק אם צדקת!

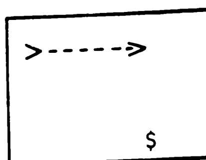
(תשובה 40 בעמוד 65)

פרק ז
יורים ברעים – משחק מסכם

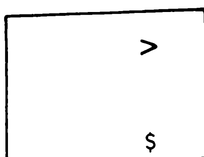
מי מאתנו לא שיחק לפחות פעם אחת באחד מאלפי משחקי-הטלויזיה?

הפעם, לשם שינוי, אתה עומד לבנות משחק כזה:

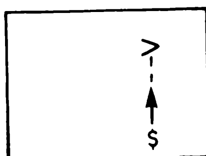
1 חללית "הרעים" (>) חולפת ללא הרף לרוחב המסך משמאל לימין. החללית שלך ($) אורבת לה למטה...



2 ברגע שנראה לך ש"הרעים" (>) נמצאים בדיוק מעל לחללית שלך, אתה לוחץ על אחד המקשים שבחרת בו להיות "הדק"...



3 ברגע זה בעצרת חללית "הרעים" והטיל שירית (↑) מתחיל לעלות מן החללית שלך ישר למעלה!



מטרת המשחק לפגוע בחללית האויב!

בוא נבדוק אם אתה כבר יודע את כל הפקודות הדרושות לבנית המשחק:

- האם אתה יודע כיצד להניע את הרעים לרוחב המסך? (כן/לא)
- האם אתה יודע לצייר את החללית שלך בתחתית המסך? (כן/לא)
- האם אתה יודע לכתוב פקודות כאלה, שהמחשב "ידע" מתי לחצת על "ההדק"? (כן/לא)
- האם אתה יודע כיצד להניע את הטיל מן החללית שלך עד למעלה? (כן/לא)

ובכן, דע לך, כי התשובה לכל אחת מארבע השאלות שבעמוד הקודם היא חיובית! בידך כל הכלים הדרושים על-מנת לכתוב תוכנית מחשב שתבצע את משחק-החלליות.

גש לעבודה

- קרא שוב בעיון את תיאור המשחק.
- בנה את התוכנית בשלבים - לפי השאלות שבעמוד הקודם.
- בדוק כל שלב על המסך ותקן בגים אפשריים.

למשל - בג אפשרי:
עם הלחיצה על "הדק" נעלמת חללית האויב - עליך להעביר את פקודת ה-IF למקום מתוחכם יותר.

(תשובה 41 בעמוד 65)

<u>הערה:</u> אם התוכנית שלך ארוכה יותר ממסך אחד, ואתה רוצה לראות חלקים ממנה - תוכל להשתמש בפקודות הבאות:

המחשב יציג את התוכנית עד שורה 1ØØ: LIST - 1ØØ •
המחשב יציג את שורות 1ØØ עד 2ØØ: LIST 1ØØ-2ØØ •
המחשב יציג את השורות מ-1ØØ עד הסוף: LIST 1ØØ - •

שכלולים אפשריים

- נסה, במקום סימן ה-$, לצייר חללית של ממש...
- לפני תחילת המשחק, המחשב שואל אותך באיזו מהירות אתה רוצה ש"הרעים" יחלפו על פני המסך.
- אחרי 1Ø "טילים" נגמר המשחק.
- המחשב ידפיס כמה "טילים" ירית ובכמה פגעת.
- חללית "רעים" שנפגעה - נופלת.

? איך "יודע" המחשב אם פגעת או לא?

רמז

החללית של "הרעים" נעה ימינה ב-FOR-NEXT. מה צריך להיות ערכו של משתנה הלולאה הזאת, כאשר הטיל שירית עולה למעלה ופוגע בחללית?

(השכלולים שלנו בתשובה 42 בעמוד 66)

שאלת חזרה לסיום

לפניך רשימת מושגים שלמדת ביחידה זו.
עבור על כל אחד מהם ובדוק אם אתה זוכר אותם.
(תוכל לרענן את זכרונך בעזרת מספרי העמודים המופיעים בסוגריים ליד כל מושג.)

FOR-NEXT	(7)	INPUT	(33)
משתנה	(10)	מחרוזת	(35)
LET	(11)	משתנה מחרוזת	(36)
LET A=A+1	(15)	שמות של משתני-מחרוזת	(37)
שמות של משתנים	(15)	המחרוזת הריקה	(38)
משתנה הלולאה	(18)	סימני פעולה במחרוזת	(40)
השהייה	(21)	LEN	(43)
:	(21)		
לולאות FOR-NEXT שתולות אחת בתוך השניה	(26)	GET	(46)
		IF ... THEN	(49)

ולסיום:

ביחידת לימוד זאת למדת להשתמש במושגים היסודיים ביותר בשפת ה-BASIC:

- FOR-NEXT.
- *משתנה מספרי ומשתנה מחרוזתי.*
- *פקודת* IF ... THEN.

יתכן שבשלב זה אתה כבר יכול להמריא על כנפי דמיונך.
אם כן, אף אחד לא מונע זאת ממך.
ייתכן שתרצה לעבור וללמוד את יחידה 3...

תשובות

הערה: התשובות מתיחסות ל-Commodore 64. בעלי VIC-20 יאלצו לבצע בתשובות אלה שינויים מעטים המתבקשים מגודל המסך השונה.

<u>תשובה 1 מעמוד 7</u>

```
10 FOR N=1 TO 8
```

<u>תשובה 2 מעמוד 7</u>

המחשב יבצע שורות אלה בדיוק 20 פעמים.

<u>תשובה 3 מעמוד 7</u>

```
                               4
10 PRINT "{CLR/HOME}{↓12}{→}";
20 FOR K=1 TO 4
                  5
30 PRINT "DANI {→}" ;
40 NEXT K
```

<u>תשובה 4 מעמוד 8</u>

```
10 PRINT "{CLR/HOME}" ;
20 FOR K=1 TO 21
30 PRINT "A{↓1}" ;
40 NEXT K
```

<u>תשובה 5 מעמוד 8</u>

יש להוסיף את השורות הבאות:

```
50 PRINT
60 PRINT "{↑2}" ;
70 FOR N=1 TO 21
80 PRINT "A{↑1}" ;
90 NEXT N
100 PRINT "{↓21}"
```

תשובה 6 מעמוד 9

```
10 PRINT "{CLR/HOME}{2→}{↓1}" ;
20 FOR L=1 TO 4
30 PRINT " ● {↓1}{2←} ▓▓▓ {↓1}{3←}{RVS-ON}
רווח → □ {1→} □ {RVS-OFF}{3→}{↑2}" ;
40 NEXT L
50 PRINT "{↓5}"
```

תשובה 7 מעמוד 14

שמות המשתנים האלה <u>אינם</u> חוקיים: 1C ,2D ,25

תשובה 8 מעמוד 16

```
10 Y=1           Y=1
20 PRINT Y       Y=1  ┌→Y=2  ┌→Y=3
30 Y=Y+1         Y=2  │ Y=3  │ Y=4
40 GO TO 20 ──────────┘ └────┘ └ - - - - →
```

תשובה 9 מעמוד 17

המספר הגדול ביותר שהמחשב כותב עם אפסים הוא 900,000,000.

תשובה 10 מעמוד 20

```
10 FOR N=1 TO 3     N=1 ┌→N=2 ┌→N=3
20 PRINT N          N=1 │ N=2 │ N=3
30 NEXT N           N=2 │ N=3 │ N=4
40 PRINT "SOF"      └───┘ └───┘   ↓
                                 SOF
```

תשובה 11 מעמוד 21

יש לשנות את שורה 20 ל:

```
20 FOR T=1 TO 2000
```

תשובה 12 מעמוד 22

```
10 PRINT "START" : FOR T=1 TO 1000 : NEXT T :
   PRINT "END"
```

תשובה 13 מעמוד 23

הסוד נעוץ בשורות

```
20 FOR T=1 TO X*X : NEXT T
30 X = X+1
```

כיוון ש-X גדל ב-1 בכל פעם, הרי שהמחשב "סופר" בתחילה רק עד 1 (1×1), אח"כ עד 4 (2×2) ואח"כ עד 9 (3×3) וכן הלאה עד מספרים גדולים מאוד.

תשובה 14 מעמוד 25

```
10 PRINT "{CLR/HOME}" ;
20 FOR S=0 TO 59
              15
30 PRINT "{HOME}{↓10}{ →}" ;
40 PRINT S ;
50 NEXT S
```

תשובה 15 מעמוד 25

```
45 FOR T=1 TO 300 : NEXT T
```

תשובה 16 מעמוד 25

```
60 GO TO 10
```

תשובה 17 מעמוד 25

```
45 FOR T=1 TO 934 : NEXT T
```

תשובה 18 מעמוד 26

השורות שנוספו או שונו הן:

```
15 FOR D=0 TO 59
40 PRINT D; ":"; S;
52 PRINT "{CLR/HOME}"
55 NEXT D
```

תשובה 19 מעמוד 26

שלילי! המחשב מתנגד נמרצות לקבל לולאות מצטלבות!

תשובה 20 מעמוד 27

קיבלת על המסך את השעון הפנימי של המחשב. שתי הספרות הימניות הן שניות, 2 האמצעיות הן דקות ושתי הספרות השמאליות ביותר הן שעות.
ברגע שאתה מדליק את המחשב שעון זה מתחיל לספור מ-Ø.
כל זמן שאינך מכבה את המחשב הוא ממשיך ועובד גם כשאינך רואה אותו על המסך. עצור את התוכנית והרץ אותה שנית, וראה כי השעון ממשיך תמיד לעבוד. כמה זמן עבר מאז הדלקת את המחשב?

תשובה 21 מעמוד 29

הערה: בתשובותינו התייחסנו לקומודור 64. אם ברשותך Vic 2Ø אז במקום 40 צ"ל 22.

□ = רווח

```
1Ø PRINT "{CLR/HOME}{↓10}"
2Ø FOR X=1 TO 4Ø
3Ø PRINT ">" ;                  ציפור
4Ø FOR T=1 TO 1ØØ : NEXT       השהייה
5Ø PRINT "{←1} □ " ;            מחיקה
6Ø NEXT X
7Ø PRINT "{↑1}" ;
8Ø GO TO 2Ø
```

כדי שהציפור לא תרד שורה - (שורה 7Ø)

תשובה 22 מעמוד 30

יש לשנות או להוסיף את השורות:

```
1Ø PRINT "{CLR/HOME}{↓10}{→5} ▒ {→20} ▒ {←21}" ;
2Ø FOR X=1 TO 19
7Ø FOR X=1 TO 19
8Ø PRINT "<";
9Ø FOR T=1 TO 1ØØ : NEXT
1ØØ PRINT "{←1} □ {←2}" ;
11Ø NEXT X
12Ø GO TO 2Ø
```

(שורות 8Ø–11Ø:) הציפור נעה שמאלה

תשובה 23 מעמוד 30

```
10 PRINT "{CLR/HOME}{↓10}{→^10 }" ;
```

ציור 4 נרות

```
20 FOR N=1 TO 4
30 PRINT "*{↓1}{←^1} ▨ {↓1}{←^1} ▨ {↓1}
   {←^2} ▨▨▨ {→^3}{↑3}" ;
40 NEXT N
```

חזרה לתחילת השורה

```
50 PRINT : PRINT "{↑1}" ;
```

הרוח נעה ימינה

```
60 FOR X=1 TO 30
70 PRINT ">" ;
80 FOR T=1 TO 100 : NEXT
90 PRINT "{←^1} □ " ;
100 NEXT X
```

□ - רווח

תשובה 24 מעמוד 32

```
10 PRINT "{CLR/HOME}{↓2}{→^4}" ;
20 FOR N=1 TO 10
30 PRINT "$" ;
40 NEXT N
```

תשובה 25 מעמוד 32

יש להוסיף את השורות:

```
15 FOR Y=1 TO 10
45 PRINT : PRINT "{→^4}" ;
50 NEXT Y
```

תשובה 26 מעמוד 32

יש להוסיף ולשנות:

```
5  LET L=20
15 FOR Y=1 TO L
20 FOR N=1 TO L
```

תשובה 27 מעמוד 34

יש להוסיף

```
3  PRINT "{CLR/HOME}" ;
60 PRINT "{HOME}" ; : GO TO 5
```

תשובה 28 מעמוד 36

```
PRINT "4*7=" ; 4*7
```

ביטוי חשבוני   מחרוזת

תשובה 29 מעמוד 37

המחשב לא יקבל את שמות המשתנים הבאים כמשתני מחרוזת:

2A$, B3, C$$, C*$

תשובה 30 מעמוד 40

כאשר המחשב שואל אותך   SIMAN = ?

הכנס לו את המחרוזת הבאה:

" ● {אדום} ● {ירוק}"

כולל מרכאות!   אל תשכח!

תשובה 31 מעמוד 41

```
10 PRINT "{CLR/HOME}"
20 INPUT L
30 PRINT "{CLR/HOME}{↓10}{15 →}" ;        למרכז המסך
40 FOR X=1 TO L-1                          צלע 1
50 PRINT "▩" ;
60 NEXT X
70 FOR X=1 TO L-1                          צלע 2
80 PRINT "▩{↓1}{1 ←}" ;
90 NEXT X
100 FOR X=1 TO L-1                         צלע 3
110 PRINT "▩{2 ←}" ;
120 NEXT X
```

המשך מעבר לדף!

המשך תשובה 31 מעמוד 41

```
       ┌130 FOR X=1 TO L-1
צלע 4  │140 PRINT " ▒{↑1}{←}" ;   (← ×1)
       └150 NEXT X
        160 PRINT "{HOME}" : GO TO 20
```

תשובה 32 מעמוד 42

```
גודל הריבוע הראשון       10 L=3
                         20 למחוק את שורה
הריבועים גדלים          155 L=L+4
להביא המסמן למקומו החדש. 160 PRINT "{←}{↑2}"; : GO TO 40   (← ×2)
```

תשובה 33 מעמוד 43

```
                  10 PRINT "{CLR/HOME}"
                  20 INPUT "NAME=" ; N$
למרכז המסך        30 PRINT "{CLR/HOME}{↓10}{→ }" ;   (→ ×10)
צלע עליונה      ┌ 40 FOR X=1 TO LEN(N$)+3
של המסגרת       └ 50 PRINT "*"; : NEXT
צלע ימנית       ┌ 60 FOR Y=1 TO 4
                └ 70 PRINT "*{↓1}{←}; : NEXT   (← ×1)
צלע תחתונה      ┌ 80 FOR X=1 TO LEN(N$)+3
                └ 90 PRINT "*{←}"; : NEXT   (← ×2)
צלע שמאלית      ┌100 FOR Y=1 TO 4
                └110 PRINT "*{↑1}{←}"; : NEXT   (← ×1)
השם עצמו         120 PRINT "{↓2}{→}"; N$   (→ ×2)
                 130 PRINT "{HOME}" : GO TO 20
```

תשובה 34 מעמוד 44

```
5 L=1
                          20
10 PRINT "{CLR/HOME}{↓10}{ →}" ;
20 FOR X=1 TO L
30 PRINT "▩" ;
40 NEXT
50 FOR X=1 TO L+1
                       1
60 PRINT "▩{↓1}{←}" ;
70 NEXT
80 FOR X=1 TO L+2
                  2
90 PRINT "▩{←}" ;
100 NEXT
110 FOR X=1 TO L+3
                       1
120 PRINT "▩{↑1}{←}" ;
130 NEXT
140 L = L+4
150 GO TO 20
```

תשובה 35 מעמוד 46

```
10 PRINT "{CLR/HOME}" ;
20 INPUT A$
30 PRINT "{HOME}{↓2}" ;
40 FOR Y=1 TO 20
50 FOR X=1 TO 20
60 PRINT A$;
70 NEXT X
80 PRINT
90 NEXT Y
100 PRINT "{HOME}" ; : GO TO 20
```

הדפסת שורה (שורות 50–70)

מספר שורות (שורות 40–90)

תשובה 36 מעמוד 49

```
10 GET A$
              1
20 PRINT "X{←}"; A$ ;
30 GO TO 10
```

תשובה 37 מעמוד 50

```
10 X=1
20 PRINT X : X = X+1
30 IF X=51 THEN END
40 GO TO 20
```

תשובה 38 מעמוד 51

```
10 PRINT "{CLR/HOME}"
20 GET A$
30 IF A$="D" THEN PRINT "DANI"
40 IF A$="O" THEN PRINT "OFER"
50 IF A$="L" THEN LIST
60 IF A$="N" THEN NEW
70 GO TO 20
```

תשובה 39 מעמוד 52

```
 3 PRINT "{CLR/HOME}"
 5 B$ = "X"
10 GET A$
15 IF A$ = "C" THEN 100
                    1
20 PRINT B$ ; "{←}" ; A$ ;
30 GO TO 10
100 GET B$
110 IF B$ = "" THEN GO TO 100
150 GO TO 10
```

חוד עיפרון ראשוני (שורה 5)

המחשב נשאר פה עד שתלחץ על מקש. (שורות 100–110)

תשובה 40 מעמוד 52

```
10 GET A$
20 IF A$ = "N" THEN PRINT
   "***" : GO TO 10
```

האם התוצאה על המסך הפתיעה אותך?

זכור: המחשב לא ממשיך "לקרוא" אחרי ה-THEN אם התנאי הכתוב ב-IF אינו מתקיים!

כלומר, המחשב בכלל לא "ראה" את פקודת GO TO 10 ולכן ריצת התוכנית נעצרה מיד.

תשובה 41 מעמוד 54

```
החללית שלנו                10 PRINT "{CLR/HOME}{↓20}{30 →}$"
גובה מעוף - "הרעים"        20 PRINT "{HOME}{↓5}"

                           30 FOR X=1 TO 40
                           40 PRINT ">" ;
                           50 FOR T=1 TO 50 : NEXT
הרעים טסים                 60 GET A$ : IF A$="□" THEN 200
                           70 PRINT "{1 ←}□" ;
שוב ושוב                   80 NEXT
                           90 PRINT "{↑1}" ;
                          100 GO TO 20

                          200 PRINT "{HOME}{↓19}{30 →}" ;
הטיל שירינו               210 FOR Y=1 TO 14
טס למעלה                  220 PRINT "↑" ;
                          230 FOR T=1 TO 50 : NEXT
                          240 PRINT "{1 ←}□{↑1}{1 ←}";
                          250 NEXT
                          260 GO TO 10
```

תשובה 42 מעמוד 54

יש לשנות או להוסיף את השורות הבאות:

2 PRINT "{CLR/HOME}"	
4 INPUT "SPEED=" ; M	באיזו מהירות?
15 PRINT "{HOME}$\{\overset{2}{\rightarrow}\}$"; N+1;"$\{\overset{10}{\rightarrow}\}$"; P	מדפיס מס' יריות ופגיעות
5Ø FOR T=1 TO M : NEXT	מהירות משתנה
2Ø5 N = N+1	מונה יריות
255 IF N=1Ø THEN END	הקצבת טילים
258 IF X=31 THEN P=P+1 : GO TO 3ØØ	מונה פגיעות

3ØØ PRINT "{↑1}" ;	
31Ø FOR I=1 TO 15	הרעים
32Ø PRINT ">" ;	נופלים
33Ø FOR T=1 TO 1ØØ : NEXT	באלכסון
34Ø PRINT "$\{\overset{1}{\leftarrow}\}$□$\{\overset{2}{\leftarrow}\}${↓1}" ;	למטה
35Ø NEXT	
36Ø GO TO 1Ø	

עד כה יצאו סדרות "מחשבת" עבור:

- מחשב COMMODORE 64
  (3 החוברות הראשונות מתאימות גם ל־VIC-20)
- מחשבי SINCLAIR:
  ZX-SPECTRUM
  ZX-81
- מחשבי APPLE
- מחשבי ATARI
- מחשב SPECTRVIDEO
- מחשב AMSTRAD
- מחשב IBM-PC
- מחשב COMMODORE 128

יצא לאור "שעשועי מחשבת־1"



"שעשועי מחשבת־1" – מציינת את תחילתה של סדרה חדשה של "מחשבת" ומיועדת לכל חובבי המחשבים הביתיים המחפשים אתגרים תכנותיים.

מה תמצא בחוברת הראשונה של "שעשועי מחשבת"?

הצעות מפורטות לפיתוח 5 משחקי מחשב המתאימים מאד לתכנות בשפת BASIC.

החוברת אינה כוללת אף תוכנית – את זאת אנו משאירים לך לפתח.

בנוסף לכך תוכל להשתתף בהגרלה נושאת פרסים של "מחשבת־BUG".

(פרטים תמצא בחוברת עצמה).



משהו על יחידה 2:

ביחידת לימוד זאת הנקראת

"ריבועי קסם"

תלמד להשתמש בכל המושגים הבסיסיים
והחשובים ביותר הקיימים בשפת ה־BASIC:

- לולאות סופיות (FOR-NEXT).
- משתנים מספריים ומחרוזתיים.
- פקודות תנאי (IF ... THEN).

נראה כיצד מושגים אלה יפתחו בפנינו עולם ומלואו;
החל בבנית שעון דיגיטלי, דרך בנית "ריבועי קסם"
וכלה בפיתוח משחק מחשב־טלויזיה.

מחשבת מ.ל. בע"מ – מערכות למידה